Building a rudimentary ARM processor using programmable logic

By: Johnny Sim, Tyler McGrew, Hyunil Lee, Cassandra Chow, Cody Harris, Louie Coyle

Contents

Introduction ARM Processor

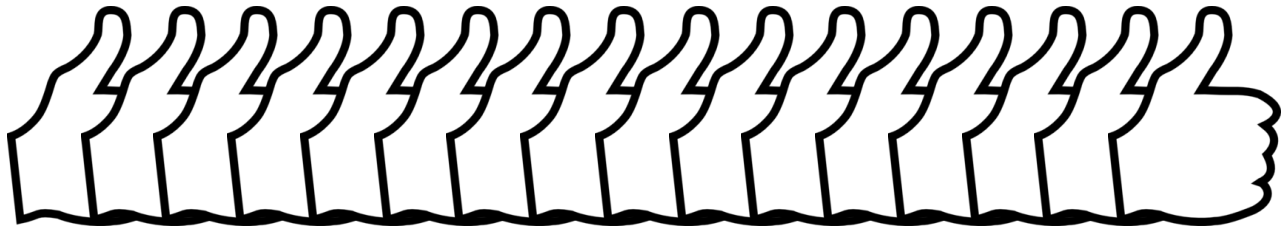
- Family of instruction set architectures for computer processors
- Simple and low cost than x86 process.
- Support 32-bit address space and 32-bit arithmetic.
- Most popular 32-bit in embedded system.

Main feature of ARM Instruction Set

- ARM Instructions are 32 bits long.
- Most instructions execute in single clock-cycle.
- Most instructions can be conditionally executed.
- A load/store architecture
- Instruction set extension via coprocessors
- Thumb 1: Very dense 16-bit compressed instruction set

Thumb Instruction Set (16-bit)

- Thumb is a 16-bit instruction set
- Core has two execution states - ARM and Thumb
- Thumb has characteristic features



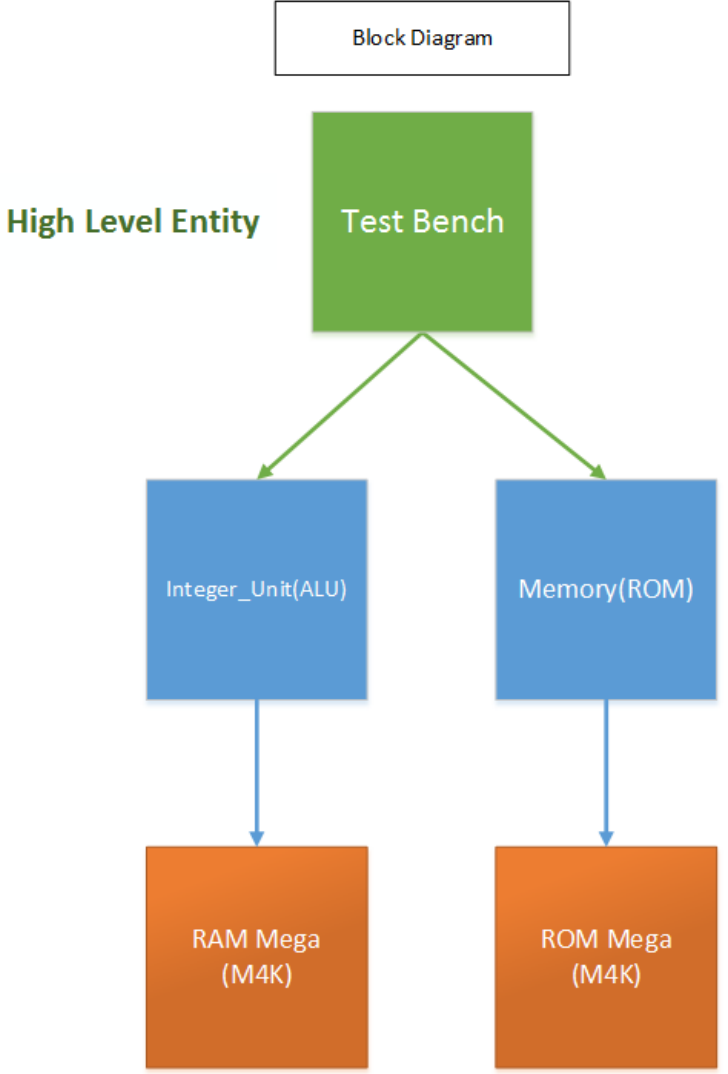
Hardware Architecture

- Three Stage Pipeline
- Harvard Architecture
- M4K Block Implementation
- Pipeline Stall/Flush Functionality
- Reset Line

Three Stage Pipeline

Block Diagram

Entity Block Diagram



Instruction Fetch Stage

- Acquire instructions from program memory
- PC: stores address of next instruction
- PC += 2 every cycle, halfword aligned
- PC LSB is ignored when addressing memory

Actual Address	Program Counter
0x00000001	0x00000002
0x00000002	0x00000004
0x00000003	0x00000006
0x00000004	0x00000008

Stall and Flush

- PC is not updated
- Instruction pipeline is cleared
- Occurs on Branch instructions

Example Instruction Sequence

	MOV R4, R0
	B subtract
	ADD R4, R1
subtract:	SUB R4, R4, R2
	MOV R0, R4

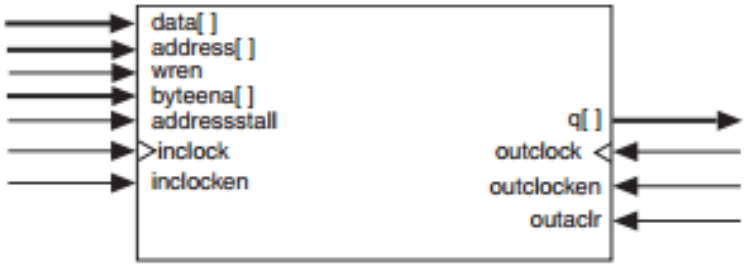
Pipeline status

Cycle	Fetch	Execute
1	MOV	0 (NOP)
2	B	MOV
3	ADD	B
4	SUB	0 (NOP)
5	MOV	SUB

Load and Store

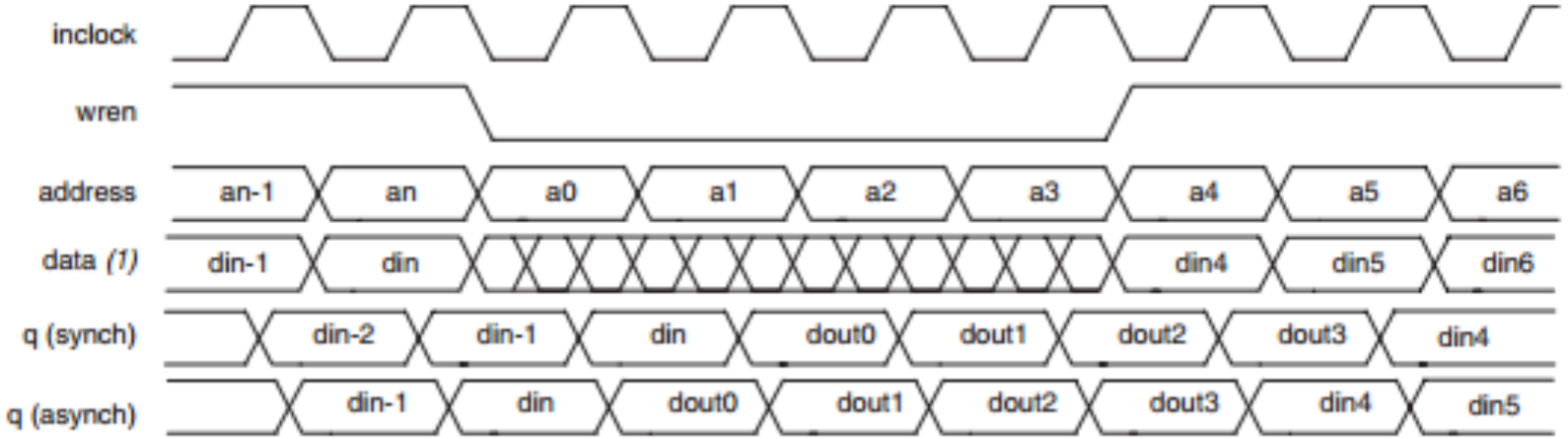
- Load/Store Architecture
- RAM instantiation
- RAM read/write access timing

Figure 8–6. Single-Port Mode *Note (1)*



Single Port RAM Timing

Figure 8–7. Cyclone II Single-Port Timing Waveforms



Note to Figure 8–7:

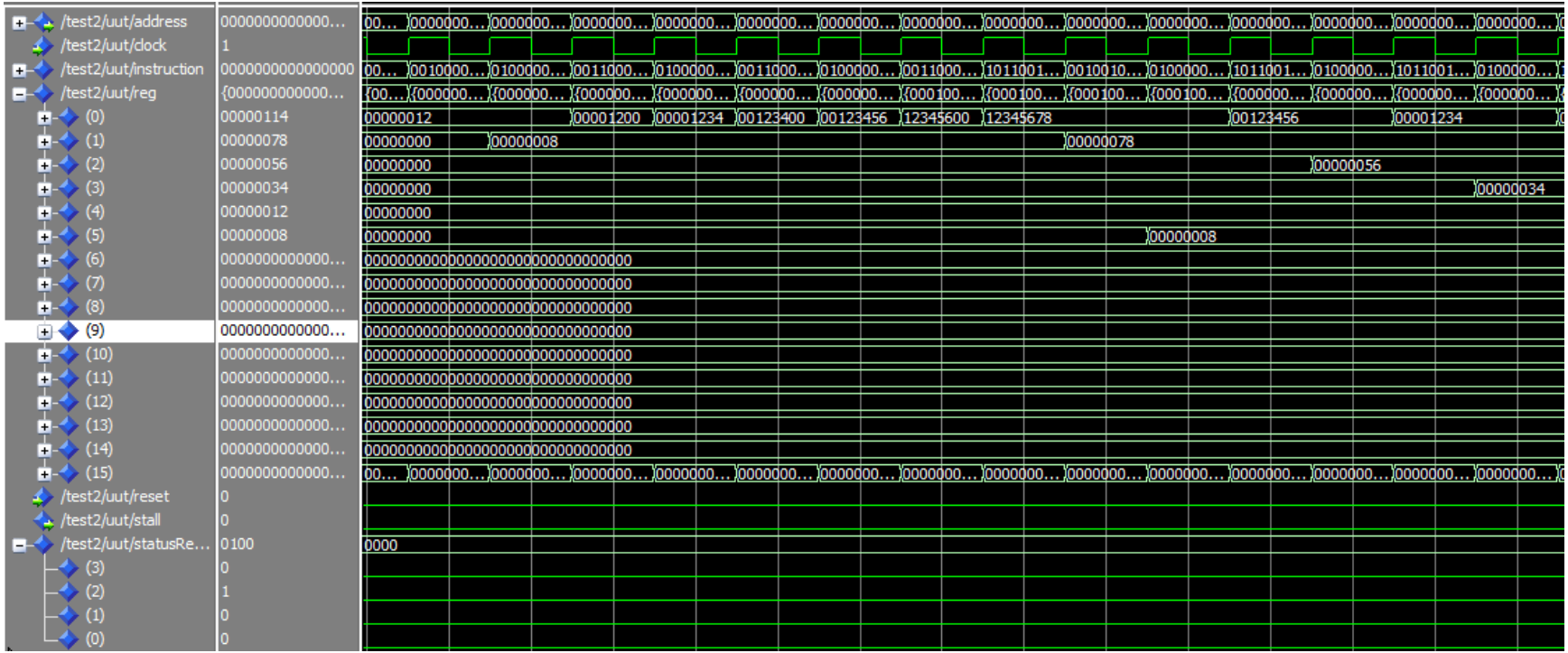
(1) The crosses in the `data` waveform during read mean "don't care."

Testing and Debugging

ModelSim

- Test Bench
- Simulation
- Waveform Viewer

Waveform Viewer



Conclusion

- Learning Arm Assembly
- Implementing 16 bit Arm processor
- Pipelining
- Building proper test bench

Thanks